

IN THE CLAIMS**Listing of Claims:**

- 1 1. (original) An apparatus for managing operations in a processor, said apparatus
2 comprising:
3 a plurality of addressable registers, each of said registers partitioned into plurality
4 of data entry fields;
5 a first comparison circuit, said first comparison circuit operable to scan and
6 compare a value in a set of said data entry fields to a predetermined input value;
7 a second comparison circuit, said second comparison circuit operable to compare
8 a first register address corresponding to a comparison match of said first comparison
9 circuit to a second register address; and
10 a dispatch circuit operable to dispatch data of a second data entry field of a
11 second register corresponding to said second register address to an operation unit in
12 response to a decode of data in a third data entry field of said second register and a
13 comparison match of said second comparison circuit.
- 1 2. (original) The apparatus of claim 1, wherein said operations are Load and Store
2 operations within said processor.
- 1 3. (original) The apparatus of claim 1, wherein said predetermined input value is a real
2 address requesting particular data corresponding to one of a Load and a Store operation.
- 1 4. (currently amended) The apparatus of claim 1, wherein said first [[scan]] comparison
2 circuit comprises multiple like entry comparison circuits, each of said multiple like entry
3 comparison circuits operable concurrently in parallel.
- 1 5. (original) The apparatus of claim 1, wherein said operation unit comprises an
2 Instruction Management Unit (IMU).

1 6. (currently amended) The apparatus of claim [[1]]2, wherein said operation unit
2 comprises a Storage Management Unit (SMU) said SMU comprising data cache memory
3 and controller and a Storage Reference Buffer (SRB).

1 7. (original) The apparatus of claim 1, wherein one of said data entry fields is a Valid
2 bit field, said Valid bit field indicating whether other data entry fields are valid.

1 8. (original) The apparatus of claim 1, wherein one of said data entry fields is an
2 Instruction Identification (ID) field corresponding to a particular Load and Store
3 operation.

1 9. (original) The apparatus of claim 1, wherein one of said data entry fields is an
2 Instruction status field corresponding to a status of one of said Load and Store operations.

1 10. (original) The apparatus of claim 1, wherein one of said data entry fields is a
2 Load/Store field having a Load/Store bit, said Load/Store bit corresponding to a Load
3 operation if said Load/Store bit has a first logic state and corresponding to a Store
4 operation if said Load/Store bit has a second logic state.

1 11. (original) The apparatus of claim of claim 1, wherein one of said data entry fields
2 comprises Real Address field, said Real Address field corresponding to a particular Real
3 Address of memory data.

1 12. (original) The apparatus of claim 1, wherein one of said data entry fields is a
2 Quadword field, said Quadword field comprising multiple bytes of data.

1 13. (original) The apparatus of claim 1, wherein one of said data entry fields is an
2 Operand Mask field, said Operand Mask field defining selected bytes of data within a
3 selected one of said data entry fields.

1 14. (original) The apparatus of claim 1, wherein said operation unit is a pipeline
2 execution unit operating concurrently on a plurality of said data entry fields.

1 15. (original) The apparatus of claim 1, wherein said addressable registers are addressed
2 using a plurality of address pointers.

1 16. (currently amended) The apparatus of claim [[1]] 6 wherein said addressable
2 registers are configured as said Storage Reference Buffer (SRB).

1 17. (original) The apparatus of claim 15, wherein one of said address pointers is a third
2 pointer, said third pointer, said third pointer pointing to one of said addressable registers
3 whose data entry fields contain data defining an earliest Store operation that is either
4 unresolved or that matches a register address of a current Load operation.

1 18. (original) The apparatus of claim 15, wherein said address pointers comprise a
2 fourth and a fifth pointer, said fourth and fifth pointers defining a window of register
3 addresses from which a Load operation may be satisfied without having to access other
4 memory storage.

1 19. (original) The apparatus of claim 15, wherein said second register address is
2 selected from registers addresses which fall within a window of register addresses, said
3 window of addresses defined by said address pointers.

1 20. (original) The apparatus of claim 15, wherein one of said address pointers is a first
2 pointer, said first pointer pointing to an IN register address of a first available register
3 into which data may be added.

1 21. (original) The apparatus of claim 15, wherein one of said address pointers is a
2 second pointer, said second pointer pointing to an OUT register address of a first
3 available register from which register data may be retired.

1 22. (original) The apparatus of claim 16, wherein said data entry fields, added to said
2 SRB after a mis-predicted branch instruction occurs in said processor, are retired and said
3 first pointer is indexed to first register address of a register with added register data entry
4 bits which were added immediately prior to said mis-predicted branch instruction.

1 23. (original) The apparatus of claim 19, wherein said window of register addresses
2 defines active Load and Store operations.

1 24. (original) The apparatus of claim 20, wherein said first pointer is indexed by one
2 when said register data has been added, said first pointer having a minimum and a
3 maximum value wherein a decrement down from a minimum value results in said
4 maximum value and an increment up from said maximum value results in said minimum
5 value.

1 25. (original) The apparatus of claim 21, wherein said second pointer is indexed by one
2 when register entry bits have been deleted, said second pointer having a minimum and
3 a maximum value wherein a decrement down from said minimum value results in said
4 maximum value and an increment up from said maximum value results in said minimum
5 value.

1 26. (original) A data processing system, comprising:

2 a central processing unit (CPU);

3 random access memory (RAM);

4 read only memory (ROM);

5 an I/O adapter; and

6 a bus system coupling devices internal to said CPU, said CPU comprising an
7 apparatus for managing operations within a processor of said CPU, said apparatus
8 comprising:

9 a plurality of addressable registers, each of said addressable registers
10 partitioned into plurality of data entry fields;

11 a first comparison circuit, said first comparison circuit operable to scan
12 and compare a predetermined input value to a value from a first data entry field selected
13 from each of said addressable registers;

14 a second comparison circuit, said second comparison circuit operable to
15 compare a first register address corresponding to a comparison match of said first
16 comparison circuit to a second register address; and

17 a dispatch circuit operable to dispatch data in a second data entry field of
18 a second register to an operation unit, said second register corresponding to said second
19 register address in response to a decode of data in a third data entry field of said second
20 register and a comparison match of said second comparison circuit.

1 27. (original) The data processing system of claim 26, wherein said operations are Load
2 and Store operations within said processor.

1 28. (original) The data processing system of claim 26, wherein said predetermined input
2 value is a real address requesting particular data corresponding to one of a Load and a
3 Store operation.

1 29. (currently amended) The data processing system of claim 26, wherein said first
2 [[scan]] comparison circuit comprises multiple like entry comparison circuits, each of
3 said multiple like entry comparison circuits operable concurrently in parallel.

1 30. (original) The data processing system of claim 26, wherein said operation unit
2 comprises an Instruction Management Unit (IMU).

1 31. (currently amended) The data processing system of claim ~~[[26]]~~27, wherein said
2 operation unit comprises a Storage Management Unit (SMU) said SMU comprising data
3 cache memory and controller and a Storage Reference Buffer (SRB).

1 32. (original) The data processing system of claim 26, wherein one of said data entry
2 fields is a Valid bit field, said Valid bit field indicating whether other data entry fields are
3 valid.

1 33. (original) The data processing system of claim 26, wherein one of said data entry
2 fields is an Instruction Identification (ID) field corresponding to a particular Load and
3 Store operation.

1 34. (original) The data processing system of claim 26, wherein one of said data entry
2 fields is an Instruction status field corresponding to a status of one of said Load and Store
3 operations.

1 35. (original) The data processing system of claim 26, wherein one of said data entry
2 fields is a Load/Store field having a Load/Store bit, said Load/Store bit corresponding to
3 a Load operation if said Load/Store bit has a first logic state and corresponding to a Store
4 operation if said Load/Store bit has a second logic state.

1 36. (original) The data processing system of claim of claim 26, wherein one of said data
2 entry fields comprises Real Address field, said Real Address field corresponding to a
3 particular Real Address of memory data.

1 37. (original) The data processing system of claim 26, wherein one of said data entry
2 fields is a Quadword field, said Quadword field comprising multiple bytes of data.

1 38. (original) The data processing system of claim 26, wherein one of said data entry
2 fields is an Operand Mask field, said Operand Mask field defining selected bytes of data
3 within a selected one of said data entry fields.

1 39. (original) The data processing system of claim 26, wherein said operation unit is a
2 pipeline execution unit operating concurrently on a plurality of said data entry fields.

1 40. (original) The data processing system of claim 26, wherein said addressable
2 registers are addressed using a plurality of address pointers.

1 41. (currently amended) The data processing system of claim ~~[[26]]~~31 wherein said
2 addressable registers are configured as said Storage Reference Buffer (SRB).

1 42. (original) The data processing system of claim 40, wherein one of said address
2 pointers is a third pointer, said third pointer, said third pointer pointing to one of said
3 addressable registers whose data entry fields contain data defining an earliest Store
4 operation that is either unresolved or that matches a register address of a current Load
5 operation.

1 43. (original) The data processing system of claim 40, wherein said address pointers
2 comprise a fourth and a fifth pointer, said fourth and fifth pointers defining a window of

3 register addresses from which a Load operation may be satisfied without having to access
4 other memory storage.

1 44. (original) The data processing system of claim 40, wherein said second register
2 address is selected from registers addresses which fall within a window of register
3 addresses, said window of addresses defined by said address pointers.

1 45. (original) The data processing system of claim 40, wherein one of said address
2 pointers is a first pointer, said first pointer pointing to an IN register address of a first
3 available register into which data may be added.

1 46. (original) The data processing system of claim 40, wherein one of said address
2 pointers is a second pointer, said second pointer pointing to an OUT register address of
3 a first available register from which register data may be retired.

1 47. (original) The data processing system of claim 41, wherein said data entry fields,
2 added to said SRB after a mis-predicted branch instruction occurs in said processor, are
3 retired and said first pointer is indexed to first register address of a register with added
4 register data entry bits which were added immediately prior to said mis-predicted branch
5 instruction.

1 48. (original) The data processing system of claim 44, wherein said window of register
2 addresses defines active Load and Store operations.

1 49. (original) The data processing system of claim 45, wherein said first pointer is
2 indexed by one when said register data has been added, said first pointer having a
3 minimum and a maximum value wherein a decrement down from a minimum value
4 results in said maximum value and an increment up from said maximum value results in
5 said minimum value.

1 50. (original) The data processing system of claim 46, wherein said second pointer is
2 indexed by one when register entry bits have been deleted, said second pointer having a
3 minimum and a maximum value wherein a decrement down from said minimum value

4 results in said maximum value and an increment up from said maximum value results in
5 said minimum value.

Claims 51-54 (canceled)

Please add the following new claims:

1 55. (new) The apparatus of claim 6, wherein the SRB comprises the plurality of
2 addressable registers.

1 56. (new) The apparatus of claim 55, wherein a Load operation in one of the Load and
2 Store operations comprises:

3 issuing concurrently a fetch instruction requesting a real address to a data cache
4 and the Storage Reference Buffer (SRB), the real address corresponding to an address of
5 multiple bytes of data;

6 scanning the addressable registers in the SRB for the real address;

7 receiving the multiple bytes of data from the SRB if the real address is available;

8 retrieving the multiple bytes of data first from the SRB if the real address is
9 available and second from the data cache if the real address is not available in the SRB;
10 and

11 updating a corresponding one of the addressable registers with the multiple bytes
12 of data.

1 57. (new) The apparatus of claim 55, wherein a Store operation in one of the Load and
2 Store operations comprises:

3 issuing a real address generation instruction;

4 looking up the real address in a table lookup buffer;

5 sending the real address to a miss resolution processor if the real address is not
6 in the table lookup buffer, the miss resolution processor determining a translated real
7 address;

8 sending the real address from one of the miss resolution processors and the table
9 lookup buffer to the SRB; and

10 updating corresponding data entry fields in one of the addressable registers in the
11 SRB.

1 58. (new) The apparatus of claim 55, wherein a Store operation in one of the Load and
2 Store operations comprises:

3 issuing an address generation instruction by a first instruction unit generating a
4 real address in memory;

5 updating the real address in a real address field of one of the addressable registers
6 in the SRB;

7 sending concurrently, a request for a multiple byte word with the real address to
8 the SRB and a data cache;

9 receiving the multiple byte word from one of the addressable registers in the
10 Storage Reference Buffer and from a data cache;

11 updating the multiple byte word from the data cache with an operand mask;

12 receiving from the first instruction unit store data operand;

13 aligning the store data operand to the multiple bytes of data; and

14 updating the multiple bytes of data with a complement of the operand mask.

1 59. (new) The apparatus of claim 55, wherein a Load operation in one of the Load and
2 Store operations comprises:

3 issuing an address generation instruction by a first instruction unit generating a
4 real address in a memory;

5 updating the real address in a real address field of one of the addressable registers
6 in the SRB;

7 sending concurrently, a request for a multiple byte word with the real address to
8 the SRB and a data cache;

9 receiving the multiple byte word from one of the addressable registers in the SRB
10 and the data cache;

11 extracting selected bytes from the multiple byte word;

12 receiving the selected bytes by the first instruction unit; and

13 updating the multiple bytes of data with a complement of the operand mask.